

UNIVERSIDAD AUTONOMA DE MADRID

ESCUELA POLITECNICA SUPERIOR



**Grado en Ingeniería de Tecnologías y Servicios de
Telecomunicación**

TRABAJO FIN DE GRADO

**DESARROLLO DE SISTEMA DE AVISOS
NUTRICIONALES ASOCIADOS CON ACTIVIDAD FÍSICA
RIGUROSA**

**Víctor García Astudillo
Tutor: Kostadin Koroutchev**

JUNIO 2018

**DESARROLLO DE SISTEMA DE AVISOS
NUTRICIONALES ASOCIADOS CON ACTIVIDAD FÍSICA
RIGUROSA**

**AUTOR: Víctor García Astudillo
TUTOR: Kostadin Koroutchev**

**Escuela Politécnica Superior
Universidad Autónoma de Madrid
Junio de 2018**

Resumen

Este Proyecto Fin de Grado trata sobre el desarrollo de una aplicación Android que sea capaz de cuantificar ejercicio físico y, una vez acabado, dar mediante notificación un aviso o consejo nutricional dependiendo del tipo de ejercicio y su intensidad.

La aplicación está dividida en 4 partes:

- La primera de ellas trata sobre perfiles. El usuario creará un perfil en el que especificará datos relevantes tales como: edad, altura, peso y sexo. A su vez estos perfiles podrán ser modificados por el usuario en cualquier momento, teniendo en cuenta de esta manera posibles cambios de peso, edad o altura (si aun está en fase de crecimiento) que el usuario pudiera experimentar.
- La segunda es la que cuantifica el ejercicio físico y es la función principal de la aplicación. Los datos arrojados serán: metros avanzados, velocidad instantánea, velocidad media, kilocalorías gastadas y el rango de frecuencia cardiaca adecuado.
- La tercera trata sobre las alarmas. El usuario podrá elegir la información que se le proporcionará durante la realización del ejercicio.
- La cuarta y última se trata de un historial. Este historial se encuentra unido al perfil del usuario y mostrará todos los ejercicios que ha realizado el usuario.

Todas estas funciones se encuentran muy conectadas entre sí y esta conexión se realiza mediante el uso de base de datos. Otra característica también usada es la de la señal GPS la cual nos servirá para detectar y contabilizar el movimiento durante el ejercicio.

La aplicación es capaz de determinar todos los datos basándose en los sensores que ya dispone el teléfono móvil.

La aplicación no se encuentra publicada en Play Store pero su apk se puede descargar desde el siguiente enlace: <https://www.dropbox.com/s/kbocj9amjuodwds/ANEF.apk?dl=0>

Palabras clave

Móvil, Android, deporte, ejercicio físico, aeróbico, anaeróbico, correr, andar, bicicleta, remo, kilocalorías, frecuencia cardiaca, avisos nutricionales.

Abstract

This Bachelor Thesis develops of an Android application that is capable to quantify physical exercise and, once finished, to give with a notification a nutritional advice depending on the type of exercise and its intensity.

The application consists of 4 parts:

- The first one is about user profiles. The user will create a profile in which he specifies relevant data like: age, height, weight and sex. The profiles may be modified at any time, taking into account possible changes in weight, age or height (if still in the growth age).
- The second part is about alarms. The user can choose the information that will be provided during the exercise.
- The third part is the one that quantifies the physical exercise and is the core function of the application. The data will be: distance, instantaneous speed, average speed, spent kilocalories and the heart rate range.
- The last one is about historial. This historial is associated to the user's profile and shows all the exercises that the user has done.

All these functions are really interrelated between each other and this connection is made by the use of database. Another feature also used is the GPS signal which helps us to detect and record movement during the exercise.

The application is able to use all the data based on the sensors available in the mobile phone.

The application is not published in Play Store but its apk can be downloaded from the following link: <https://www.dropbox.com/s/kbocj9amjuodwds/ANEF.apk?dl=0>

Keywords

Mobile, Android, sports, physical exercise, aerobic, anaerobic, running, walking, cycling, rowing, kilocalories, heart rate, nutritional warnings.

Agradecimientos

A mi familia. Por su paciencia, por ser un apoyo incondicional todo este tiempo, por creer en mí cuando ni yo mismo lo hacía, por celebrar cada victoria y por apoyarme en cada derrota.

A mis amigos. Por ser esa familia que uno elige y por todos esos momentos de risas y desconexión que hacían que los problemas y agobios de la universidad se vieran lejanos y pequeños.

A mis compañeros de carrera. Por todas esas tardes de estudio, por esas explicaciones que valían mil tutorías, por la compañía en los viajes de tren o autobús, por las charlas surrealistas en las comidas y por mil momentos más que conseguían que incluso los días con largas clases, practicas interminables, agobiantes exámenes o con todo lo anterior junto acabasen con un gran sabor de boca.

Y a mi tutor. Por darme la oportunidad de realizar este proyecto y estar siempre disponible.

Porque sin todos vosotros no se habría logrado.

INDICE DE CONTENIDOS

1	Introducción.....	1
1.1	Motivación.....	1
1.2	Objetivos.....	1
1.3	Organización de la memoria.....	2
2	Estado del arte.....	3
2.1	Endomondo.....	3
2.2	RunKeeper.....	3
2.3	Google fit.....	4
2.4	Strava.....	4
2.5	Resumen.....	4
3	Diseño.....	5
3.1	Requisitos.....	5
3.1.1	Elección del Sistema Operativo.....	5
3.1.1.1	Comparación de popularidad.....	5
3.1.1.2	Comparación de evolución en ventas.....	6
3.1.1.3	Comparación general.....	6
3.1.1.4	Comparación de recursos.....	6
3.1.1.5	Decisión.....	7
3.1.2	Versión Sistema Operativo.....	7
3.1.3	Pantallas.....	8
3.2	Diagrama de bloques.....	10
3.3	Estructura de la aplicación.....	11
3.3.1	Actividades.....	11
3.3.2	Scripts auxiliares.....	11
3.3.3	Base de datos.....	11
3.3.4	Servicios.....	12
4	Desarrollo.....	13
4.1	Primeros pasos.....	13
4.2	Herramientas.....	13
4.3	Partes de una aplicación.....	13
4.3.1	Carpeta manifests.....	14
4.3.2	Carpeta java.....	14
4.3.3	Carpeta resources.....	16
4.3.4	Carpeta Gradle Scripts.....	17
4.4	Partes de la aplicación.....	18
4.4.1	Pantalla principal.....	18
4.4.2	Pantalla perfiles.....	19
4.4.2.1	Pantalla añadir perfil.....	20
4.4.2.2	Pantalla modificar perfil.....	21
4.4.2.3	Pantalla cargar perfil.....	23
4.4.3	Pantalla ejercicio.....	24
4.4.4	Pantalla alarmas.....	28
4.4.5	Pantalla historial.....	30
5	Integración, pruebas y resultados.....	31
5.1	Integración.....	31
5.2	Pruebas y resultados.....	32
5.2.1	Prueba 1.....	32
5.2.1.1	Perfil.....	32
5.2.1.2	Ejercicio aeróbico.....	33
5.2.1.3	Ejercicio anaeróbico.....	34

5.2.1.4 Historial.....	35
5.2.2 Prueba 2.....	36
5.2.2.1 Perfil.....	36
5.2.2.2 Ejercicio aeróbico.....	36
5.2.2.3 Ejercicio anaeróbico.....	37
5.2.2.4 Historial.....	38
6 Conclusiones y trabajo futuro.....	39
6.1 Conclusiones.....	39
6.2 Trabajo futuro.....	40
Referencias.....	41
Glosario.....	- 1 -

INDICE DE FIGURAS

FIGURA 1: LOGO ENDOMONDO.....	3
FIGURA 2: LOGO RUNKEEPER.....	3
FIGURA 3: LOGO GOOGLE FIT.....	4
FIGURA 4: LOGO STRAVA.....	4
FIGURA 5: VENTAS S.O. MÓVILES.....	5
FIGURA 6: VERSIONES ANDROID.....	8
FIGURA 7: GRÁFICO USO VERSIONES ANDROID.....	8
FIGURA 8: TAMAÑOS PANTALLAS ANDROID.....	9
FIGURA 9: TAMAÑOS DE PANTALLAS MÓVILES MÁS FABRICADOS (2014).....	9
FIGURA 10: TAMAÑOS DE PANTALLAS MÓVILES MÁS FABRICADOS (2016).....	9
FIGURA 11: DIAGRAMA DE BLOQUES.....	10
FIGURA 12: PARTES DE UNA APLICACIÓN.....	14
FIGURA 13: CICLO DE VIDA DE UNA ACTIVIDAD.....	15
FIGURA 14: PANTALLA PRINCIPAL.....	18
FIGURA 15: PANTALLA PERFILES.....	19
FIGURA 16: PANTALLA AÑADIR PERFIL.....	20
FIGURA 17: PANTALLA MODIFICAR PERFIL.....	21
FIGURA 18: PANTALLA CARGAR PERFIL.....	23
FIGURA 20: PANTALLA EJERCICIO.....	24
FIGURA 19: PANTALLA ALARMAS.....	28
FIGURA 21: PANTALLA HISTORIAL.....	30
FIGURA 22: INTEGRACIÓN EN DISPOSITIVO MÓVIL DE 3.5 PULGADAS VS INTEGRACIÓN EN DISPOSITIVO MÓVIL DE 5 PULGADAS.....	32
FIGURA 23: PRUEBA 1. PERFIL.....	32
FIGURA 24: PRUEBA 1. EJERCICIO AERÓBICO.....	33

FIGURA 25: PRUEBA 1. NOTIFICACIÓN DEL EJERCICIO AERÓBICO.....	33
FIGURA 26: PRUEBA 1. EJERCICIO ANAERÓBICO.....	34
FIGURA 27: PRUEBA 1. NOTIFICACIÓN DEL EJERCICIO ANAERÓBICO.....	34
FIGURA 28: PRUEBA 1. HISTORIAL PARTE 1.....	35
FIGURA 29: PRUEBA 1. HISTORIAL PARTE 2.....	35
FIGURA 30: PRUEBA 2. PERFIL.....	36
FIGURA 31: PRUEBA 2. EJERCICIO AERÓBICO.....	36
FIGURA 32: PRUEBA 2. NOTIFICACIÓN DEL EJERCICIO AERÓBICO.....	36
FIGURA 33: PRUEBA 2. EJERCICIO ANAERÓBICO.....	37
FIGURA 34: PRUEBA 2. NOTIFICACIÓN DEL EJERCICIO ANAERÓBICO.....	37
FIGURA 35: PRUEBA 2. HISTORIAL PARTE 1.....	38
Figura 36: Prueba 2. Historial parte 2.....	38

INDICE DE TABLAS

TABLA 1: RESUMEN ESTADO DEL ARTE.....	4
TABLA 2: SUBCARPETAS DE LA CARPETA RESOURCES.....	17
Tabla 3: Tabla valores METs.....	26

1 Introducción

1.1 Motivación

Este TFG está particularmente orientado a aquellos usuarios que disfruten del ejercicio físico, ya sea de manera habitual o esporádica y que deseen obtener más información sobre el ejercicio que realicen.

Este TFG surge por tanto como una herramienta para una vida más saludable, que permita a sus usuarios, gracias a la cuantificación de ejercicio físico, a los avisos nutricionales y a la personalización por parte de los perfiles, mantener un registro de todos los ejercicios que realice.

Los aspectos técnicos desarrollados ya se encuentran disponibles en otras aplicaciones, por tanto la idea de esta aplicación es la de ofrecer una versión mejorada de los mismos, consistente en el uso de avisos nutricionales dependiendo del ejercicio y de la ubicación para la estimación de la distancia recorrida y no del uso del podómetro.

1.2 Objetivos

El objetivo personal y educacional es aprender sobre programación Android, ya que es un tema que forma una parte esencial en la tecnología actual y vi por tanto en este TFG la oportunidad perfecta para poder aprender sobre ello.

El objetivo para el usuario sería que aprendiese como cuantificar el ejercicio y motivarle a superarse mediante avisos nutricionales que le aportaran información útil y mediante el historial para que pudiera ver en una misma tabla todos los ejercicios realizados con los tiempos, distancia, velocidad y kilocalorías pudiendo ver así su mejora.

El objetivo como programador era el de conseguir crear la base de datos con éxito, comunicarse con ella sin problema desde cualquier parte de la aplicación y obtener las características del ejercicio físico de la forma más precisa posible. En un principio se intentó la conexión bluetooth con una pulsera de actividad para la recopilación de datos del ejercicio físico, pero finalmente por falta de tiempo y recursos se optó por usar los propios sensores del móvil, en específico, el GPS ya que era el que más precisión nos aportaba.

1.3 Organización de la memoria

La memoria consta de los siguientes capítulos:

- I. Introducción
- II. Estado del arte
- III. Diseño
- IV. Desarrollo
- V. Integración, pruebas y resultados
- VI. Conclusiones y trabajo futuro

2 Estado del arte

Las aplicaciones deportivas están en auge, cada vez más gente se preocupa por mantener una vida más saludable y de mantener una cierta actividad física de la forma más regular posible, es por ello que el número de aplicaciones existentes que se dedican a cuantificar el ejercicio realizado está aumentando a su vez.

A continuación podremos ver un número de aplicaciones (las más usadas) que se dedican a ello junto a una breve descripción y finalmente una pequeña comparativa entre ellas y la aplicación desarrollada en este TFG.

2.1 Endomondo



Figura 1: Logo Endomondo

- Esta aplicación cuenta con más de 10 millones de descargas en Play Store.
- Posee un registro de actividad basado en señal GPS e incluso es capaz de trabajar sin él (aunque es menos preciso).
- No tiene la posibilidad de conectarse a una pulsera de actividad.
- Funciona a base de perfiles y es necesario tener uno para poder usar la aplicación.
- No dispone de ningún tipo de aviso nutricional al acabar el ejercicio.
- Posee un historial de actividad intuitivo y de fácil manejo.

2.2 RunKeeper



Figura 2: Logo RunKeeper

- Esta aplicación cuenta con más de 10 millones de descargas en Play Store.
- Posee un registro de actividad basado en señal GPS que no es muy preciso y que da algún que otro problema de ubicación que afecta a los cálculos de distancia.
- No tiene la posibilidad de conectarse a una pulsera de actividad.
- Funciona a base de perfiles y es necesario tener uno para poder usar la aplicación.
- No dispone de ningún tipo de aviso nutricional al acabar el ejercicio.
- Posee un historial de actividad intuitivo y de fácil manejo.

2.3 Google fit



Figura 3: Logo Google fit

- Esta aplicación cuenta con más de 10 millones de descargas en Play Store.
- Posee un registro de actividad basado en señal GPS que es bastante preciso.
- Tiene la posibilidad de conectarse a una pulsera de actividad y poder leer los datos a través de ella, no posee por tanto una dependencia de llevar el móvil tan alta ya que las pulseras de actividad son más fáciles de transportar.
- No funciona a base de perfiles, únicamente posee un dato asociado al peso, la altura y la edad que puede ser cambiado en cualquier momento.
- No dispone de ningún tipo de aviso nutricional al acabar el ejercicio.
- Posee un historial de actividad intuitivo y de fácil manejo.

2.4 Strava



Figura 4: Logo Strava

- Esta aplicación cuenta con más de 10 millones de descargas en Play Store.
- Posee un registro de actividad basado en señal GPS que es bastante preciso.
- No tiene la posibilidad de conectarse a una pulsera de actividad.
- Funciona a base de perfiles y es necesario tener uno para poder usar la aplicación.
- No dispone de ningún tipo de aviso nutricional al acabar el ejercicio.
- Posee un historial de actividad no intuitivo. Posee una organización a base de gráficas que la hace difícil de entender en un primer uso.

2.5 Resumen

Aplicaciones	Registro de actividad correcto	Conexión con pulsera	Perfiles	Avisos nutricionales	Historial intuitivo
Endomondo	Sí	No	Sí	No	Sí
RunKeeper	No	No	Sí	No	Sí
Google fit	Sí	Sí	No	No	Sí
Strava	Sí	No	Sí	No	No
TFG (ANEF)	Sí	No	Sí	Sí	Sí

Tabla 1: Resumen Estado del arte

3 Diseño

3.1 Requisitos

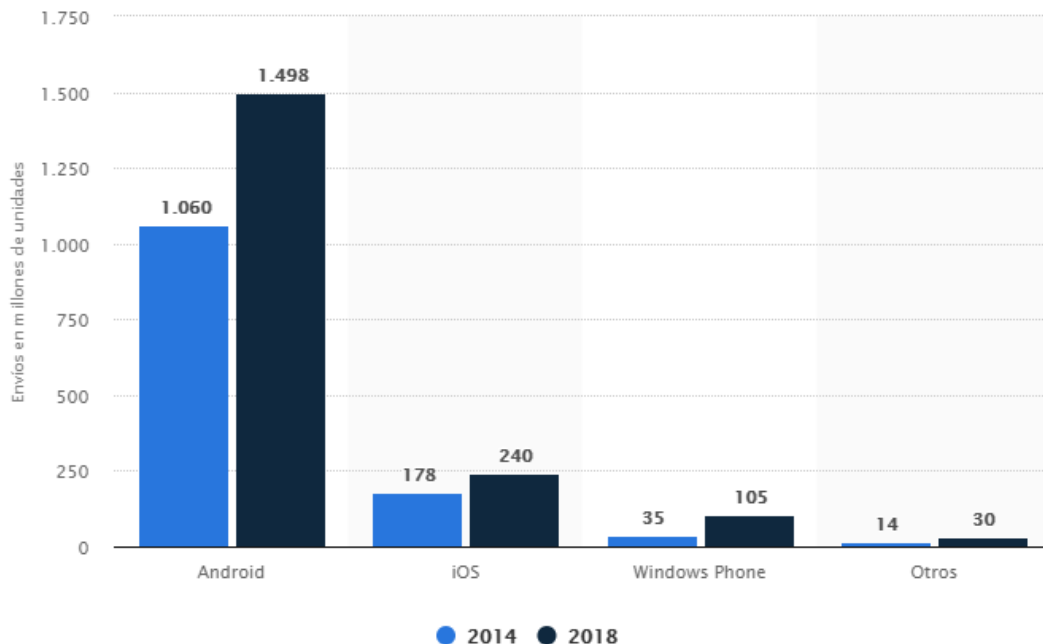
A continuación se expondrán todos los requisitos que se tuvieron en cuenta para la realización de la aplicación.

3.1.1 Elección del Sistema Operativo

En un primer paso se decidió qué SO iba a ser usado, para ello se analizaron varias características de todos ellos para poder determinar cuál iba a ser el utilizado.

3.1.1.1 Comparación de popularidad

Cabe destacar que la popularidad no es el factor más importante puesto que un sistema operativo popular entre los usuarios no garantiza nada para los desarrolladores ya que puede ocurrir que el peor sistema operativo en términos de desarrollo sea el más popular entre los usuarios. Pero aun así tiene cierta importancia ya que el número de clientes potenciales a los que se puede acceder va a ser mayor si usamos un sistema operativo que sea muy usado. Dicho esto a continuación se mostrará una figura que muestra las ventas de dispositivos en el año 2014 y 2018 clasificados por su sistema operativo:



© Statista 2018

Figura 5: Ventas S.O. móviles

Como se puede ver existen dos sistemas operativos claramente superiores al resto, estos son IOS y Android y de entre los demás el más usado, aunque por poco, es Windows Phone.

El resto del análisis por tanto lo haremos sobre estos 3 sistemas operativos ya que, como ya se ha dicho, son los que mayor cantidad de clientes potenciales nos ofrece.

3.1.1.2 Comparación de evolución en ventas

Otro tema importante es la evolución a lo largo de los años ya que la popularidad de cualquier cosa puede ser finalmente solo moda y que por tanto en un espacio bastante corto de tiempo se convierta en algo obsoleto. Esto, por supuesto, es también aplicable a los sistemas operativos.

En la figura anterior (ver Figura 5: Ventas S.O. móviles) se puede observar como en un espacio de 4 años la mayor evolución de mercado la tuvieron tanto IOS como Android, especialmente este último, intuyendo por tanto una evolución similar en los próximos años.

3.1.1.3 Comparación general

Windows Phone: El sistema operativo de los dispositivos Microsoft.

- Como punto a favor tiene la sincronización ya que, al tener el mismo sistema operativo que la mayoría de los ordenadores existentes, ofrece una conectividad muy eficiente. Otro punto a favor es su solidez, ya que es poco frecuente que alguna aplicación se bloquee.
- Como punto en contra tiene el limitado catalogo de aplicaciones que ofrece y es que incluso algunas de las aplicaciones más conocidas no tienen una versión para este sistema operativo. Además posee una limitada capacidad multitarea.

IOS: El sistema operativo de Apple.

- Como punto a favor tiene la calidad de sus aplicaciones, la ausencia de virus y la velocidad de sus actualizaciones.
- Como punto en contra se encuentra la exclusividad de estos dispositivos y las limitaciones que ofrecen al usuario a la hora de usarlo, ya que no permite acceder o personalizar ciertas funciones que otros modelos de móviles (incluso mucho más antiguos) sí permiten.

Android: Desarrollado por Google, su implementación no se encuentra limitado a ninguna marca o compañía.

- Como punto a favor tiene su sencillez, su variedad (ya que multitud de dispositivos utilizan este sistema operativo), la personalización casi ilimitada que puede ofrecer así como el hecho de que las aplicaciones sean de código abierto.
- Como punto en contra tiene la batería, ya que se consume con mucha rapidez y la lentitud de las actualizaciones.

3.1.1.4 Comparación de recursos

Para un desarrollador de aplicaciones los recursos ofrecidos por cada sistema operativo es un tema a tener muy en cuenta ya que si estos son muy limitados entorpecerán la labor del desarrollador llegando incluso a hacerla inviable. De estas tres aplicaciones que se han analizado en los puntos anteriores las más limitadas son Windows Phone e IOS y la menos limitada es Android.

En el caso de **Windows Phone** las limitaciones se encuentran en su bajo desarrollo, al ser un sistema operativo poco común el número de herramientas para desarrollar son muy limitadas. Su coste de desarrollo total estaría sobre los 1500\$.

IOS ofrece limitaciones tanto en su desarrollo, ya que las herramientas son todas de pago y no son multiplataforma, lo que obligaría a utilizar ordenadores Mac para ello, como en lo económico y es que una cuenta de desarrollador cuesta 99\$ mensuales. Su coste de desarrollo total estaría sobre los 3000 \$.

Android sin embargo no ofrece ninguna de las limitaciones anteriores, sus herramientas de desarrollo son totalmente gratuitas y además multiplataforma por lo que cualquier sistema operativo de ordenador permitiría su desarrollo. Su precio también es un punto a favor ya que la cuenta de desarrollador únicamente necesita un solo pago de 25\$. Su coste de desarrollo total estaría sobre los 300\$.

3.1.1.5 Decisión

Finalmente, tras realizar este análisis, se decidió usar como sistema operativo el Sistema **Android** debido a que, como se ha visto en los análisis anteriores, es el sistema operativo más adecuada debido a su sencillez, su bajo precio de desarrollo y a que es la opción que menos tiempo necesitaría dedicar a recursos pudiendo emplearlo por tanto en la realización de la aplicación.

3.1.2 Versión Sistema Operativo

Una vez elegido el sistema operativo más idóneo para realizar la aplicación faltaría saber qué versiones del mismo serán compatibles con nuestra aplicación.

Cada versión de Android tiene asociado un nivel de API, este nivel de API se encuentra directamente relacionado con las características que se pueden utilizar. Por tanto, cuanto mayor sea el nivel de API mayor será el número de características del móvil al que podremos acceder en nuestro desarrollo de la aplicación. Un ejemplo de esto sería el podómetro de los móviles, para los niveles bajos de API es inaccesible pero para los niveles altos (a partir del 19) sí es accesible y por tanto programable.

A continuación se muestra la siguiente figura donde se puede ver cada versión de Android con su correspondiente nivel de API:

Version	Codename	API	Distribution
2.3.3 - 2.3.7	Gingerbread	10	0.3%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	0.4%
4.1.x	Jelly Bean	16	1.5%
4.2.x		17	2.2%
4.3		18	0.6%
4.4	KitKat	19	10.3%
5.0	Lollipop	21	4.8%
5.1		22	17.6%
6.0	Marshmallow	23	25.5%
7.0	Nougat	24	22.9%
7.1		25	8.2%
8.0	Oreo	26	4.9%
8.1		27	0.8%

Figura 6: Versiones Android

Por tanto, elegir un nivel de API muy grande implicaría una reducción de accesibilidad por parte de los clientes, ya que estos niveles de API solo se encuentran en los últimos y más caros modelos. En el siguiente gráfico se muestra de manera proporcional cuanto se usa cada versión de Android:

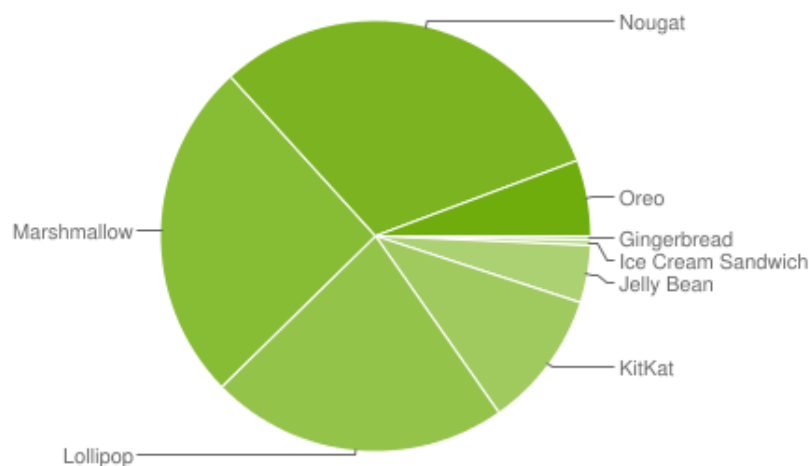


Figura 7: Gráfico uso versiones Android

Finalmente se decidió que la aplicación estuviera disponible para todos los dispositivos Android a partir de la API 16 porque así se podía llegar al mayor número de usuarios posible y porque las características que se necesitaban para poder desarrollar la aplicación no necesitaba de niveles de API más altos que ese.

3.1.3 Pantallas

Una de las características del Sistema operativo Android es su diversidad, tiene un gran número de modelos de dispositivos que abarcan desde móviles a tablets e incluso a smartwatch.

En esta variedad de modelos también existe una gran variedad de pantallas, ya que no todos los modelos tienen las mismas dimensiones. Android divide el tamaño de sus pantallas en 4: *small*, *normal*, *large* y *xlarge*.

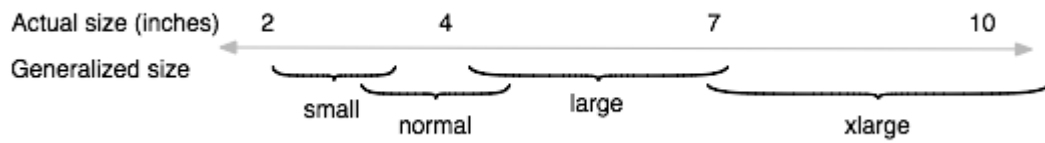


Figura 8: Tamaños pantallas Android

Entre toda esta variedad es importante saber cuál es la opción más popular, ya que el tamaño más popular de tamaño nos arrojará un dato muy importante sobre cuál es el tamaño a tener en cuenta al desarrollar nuestra aplicación. En las siguientes gráficas veremos una clasificación por tamaño de pantalla de los móviles fabricados en los años 2014 y 2016:

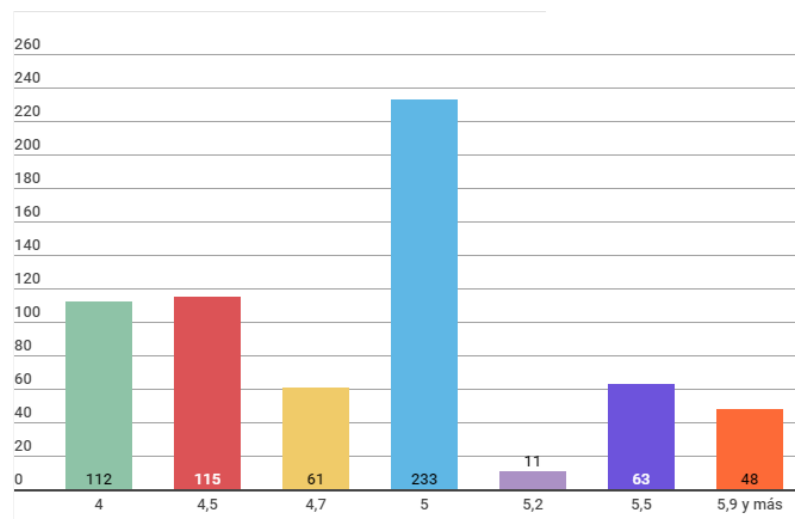


Figura 9: Tamaños de pantallas móviles más fabricados (2014)

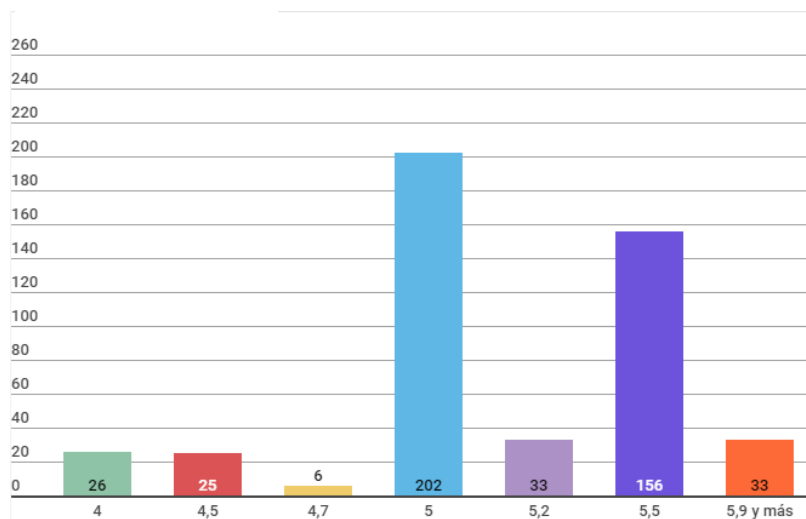


Figura 10: Tamaños de pantallas móviles más fabricados (2016)

Tal y como se puede observar en las gráficas los móviles más fabricados (Y por tanto se intuye, los que más ingresos dan) son los de 5 pulgadas mientras que los pequeños (tamaño *small*) y los más grandes (tamaño *xlarge*) han visto reducida su producción. De esta forma los móviles más comunes estarían entre los tamaños *normal* y *large* que son los cercanos a las 5 pulgadas.

Por tanto, la aplicación ha sido desarrollada teniendo en cuenta este tamaño de pantalla (5 pulgadas) ya que pese a que para los ejercicios físicos sean más populares los dispositivos pequeños, esta aplicación al ser pensada para un teléfono móvil se intenta adaptar a las preferencias de los usuarios de los teléfonos móviles. Aun así, cabe destacar que el propio sistema Android es capaz de adaptar el tamaño de los elementos que forman la aplicación a cualquiera de los tamaños de pantalla del móvil en el que se esté ejecutando, de esta forma, aunque una aplicación no haya sido diseñada específicamente para un tamaño de móvil ésta se podrá adaptar. Cabe destacar también que esta adaptación no es perfecta y en los móviles más grandes podrían llegar a notarse los elementos muy separados y en los más pequeños muy juntos o incluso cortados.

3.2 Diagrama de bloques

La aplicación ha sido diseñada de la forma más sencilla posible de forma que el usuario aprenda a usarla con rapidez. De esta forma podemos decir que su usabilidad y eficiencia es completamente adecuada.

A continuación se muestra mediante un diagrama de bloques la distribución de la aplicación:

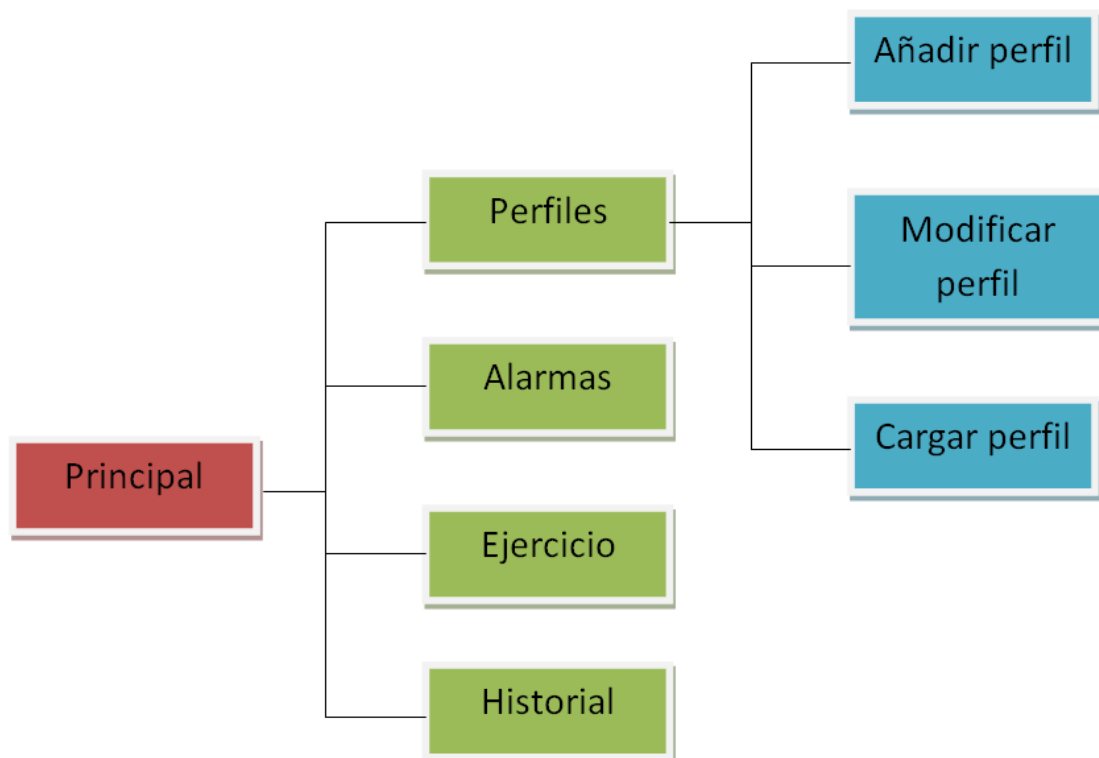


Figura 11: Diagrama de bloques

3.3 Estructura de la aplicación

3.3.1 Actividades

La aplicación se encuentra formada por un total de 8 actividades, cada una de estas 8 actividades está formada a su vez por un archivo java y un archivo xml. Por lo que, en realidad, las 8 actividades forman un total de 16 ficheros.

Estas 8 actividades ya mencionadas se encuentran divididas en 3 niveles (Como se puede ver en el diagrama de bloques del apartado anterior) cada uno de estos tres niveles cumple una función cada vez más específica según va aumentando el nivel. Estos niveles son:

Primer nivel: Este primer nivel es el más simple de todos, abarca únicamente una actividad y se trata del menú de bienvenida de la aplicación, todas las actividades son accesibles desde este nivel.

Segundo nivel: En este segundo nivel es donde se encuentran todas las funciones de la aplicación, este nivel es el que se encarga de todas las operaciones relacionadas con la función elegida en el nivel anterior, este nivel tiene acceso al primer nivel pero no tiene acceso a ninguna de las otras funciones de su mismo nivel. A su vez, solo una parte específica de este nivel accede al tercer y último nivel.

Tercer nivel: Este es el nivel más específico de todos, desde el mismo solamente se puede acceder al nivel 2 y su función es la de ampliar las funciones de un elemento del nivel anterior.

3.3.2 Scripts auxiliares

A diferencia de las actividades los scripts de ayuda únicamente están formados por un archivo java y este ayudará en el funcionamiento de las actividades.

Solamente hay un script de ayuda llamado *Table* y que ayudará a la actividad historial. Su función es la de crear y editar una tabla en historial mediante el uso de comandos simples sin necesidad de realizar muchas líneas de código.

3.3.3 Base de datos

Las bases de datos son fundamentales en la aplicación, ya que además de guardar los datos que arroja el usuario permite conectar todas las actividades y hacer que lo que se haga en una de ellas pueda tener repercusión en las demás.

La base de datos de la aplicación se encuentra dividida en las siguientes partes:

Base datos perfiles: En ella se guardarán todos los perfiles creados.

Base datos alarmas: En ella se guardará la configuración de alarmas que el usuario ha elegido para ser notificado.

Base datos historial: En ella se guardarán todos los datos de las actividades físicas realizadas.

Base datos load: En ella se guardarán los datos del perfil cargado por el usuario.

3.3.4 Servicios

Los servicios nos permiten realizar actividades en segundo plano, es decir, nos permiten ejecutar dos acciones a la vez. En el caso de esta aplicación su uso era muy necesario ya que para los avisos nutricionales era necesario un tiempo de espera de 10 minutos, el cual si no se realizara en paralelo ralentizaría toda la aplicación.

4 Desarrollo

A continuación se expondrá todo el desarrollo llevado a cabo para la realización de esta aplicación.

4.1 Primeros pasos

En un primer lugar se realizaron tutoriales de YouTube y de otras páginas de internet con el fin de adquirir los conocimientos necesarios del tema en cuestión.

Gracias a los conocimientos sobre java que ya se tenían los primeros pasos no fueron tan largos ni costosos de realizar y en poco tiempo se pudo empezar a programar pequeñas tareas que al final ayudarían a la comprensión de tareas más complejas como pueden ser la creación de bases de datos y su interacción con las mismas o el uso de sensores presentes en el dispositivo móvil.

4.2 Herramientas

Para el desarrollo de esta aplicación se ha utilizada el programa Android Studio el cual es de acceso libre y se puede descargar desde la página oficial de Android. Las simulaciones se han realizado sobre dos dispositivos móviles físicos con distintas versiones de Android (4.1 y 7.0).

Estas simulaciones se podrían realizar a través del simulador de Android Studio, el cual permite elegir la versión de Android y hasta el tamaño de pantalla, pero por motivos de rapidez en simulación y con la intención de poder probar la aplicación en el exterior estas simulaciones se decidieron hacer sobre los dispositivos físicos antes mencionados.

4.3 Partes de una aplicación

Una aplicación se divide en dos grandes carpetas, estas carpetas contienen todos los elementos que componen y hacen funcionar la aplicación. Estas dos carpetas son *app* y *Gradle Scripts*.

Sin embargo, de entre estas dos carpetas cobra algo más de importancia la carpeta *app*, la cual se divide a su vez en 3 subcarpetas, como se puede ver en la siguiente figura:

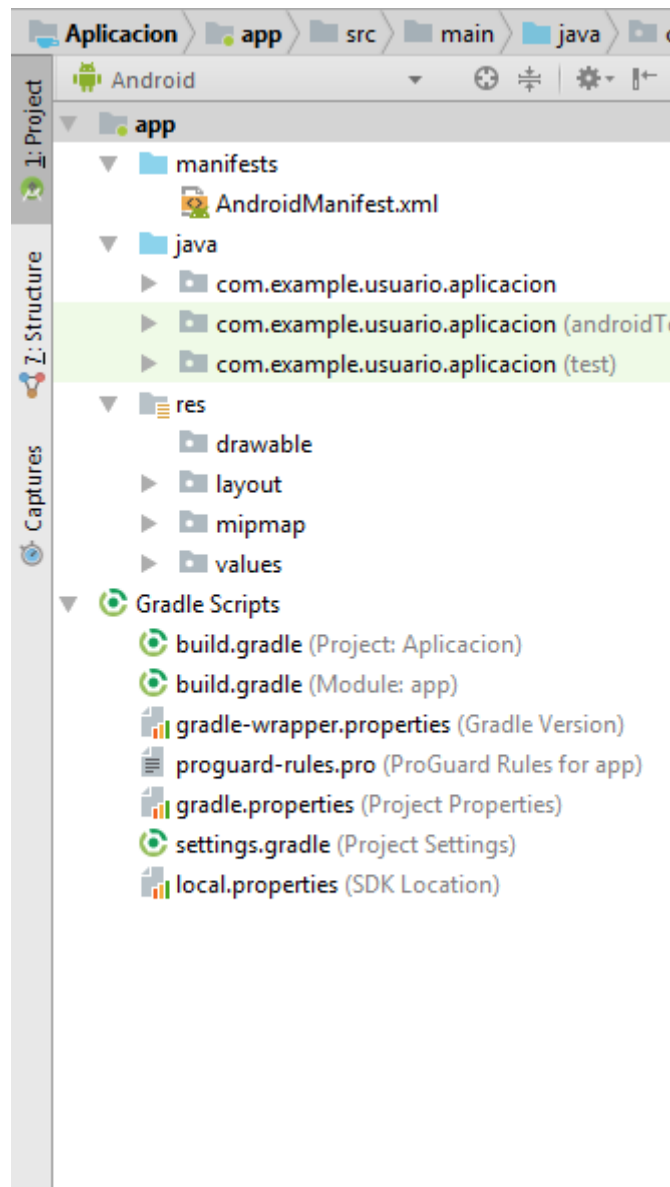


Figura 12: Partes de una aplicación

A continuación se explicarán cada una de las 4 carpetas que se ven desplegadas en la figura.

4.3.1 Carpeta manifests

Contiene un fichero xml en el que se definen muchos de los aspectos principales de la aplicación como pueden ser su identificación (nombres, iconos...), sus componentes o los permisos necesarios para su funcionamiento.

4.3.2 Carpeta java

Esta carpeta contendrá el código fuente completo de la aplicación, este código fuente puede pertenecer a una actividad (formada por un archivo java y un archivo xml) o ser un script auxiliar que ayude a alguna actividad. Estos archivos poseen lo que se denomina como el "ciclo de vida de una actividad", este ciclo de vida se compone de una serie de métodos que definen su forma de actuar. En la siguiente figura se puede observar cómo

funciona este ya mencionado ciclo de vida y le seguirá una breve explicación de cada uno de los métodos:

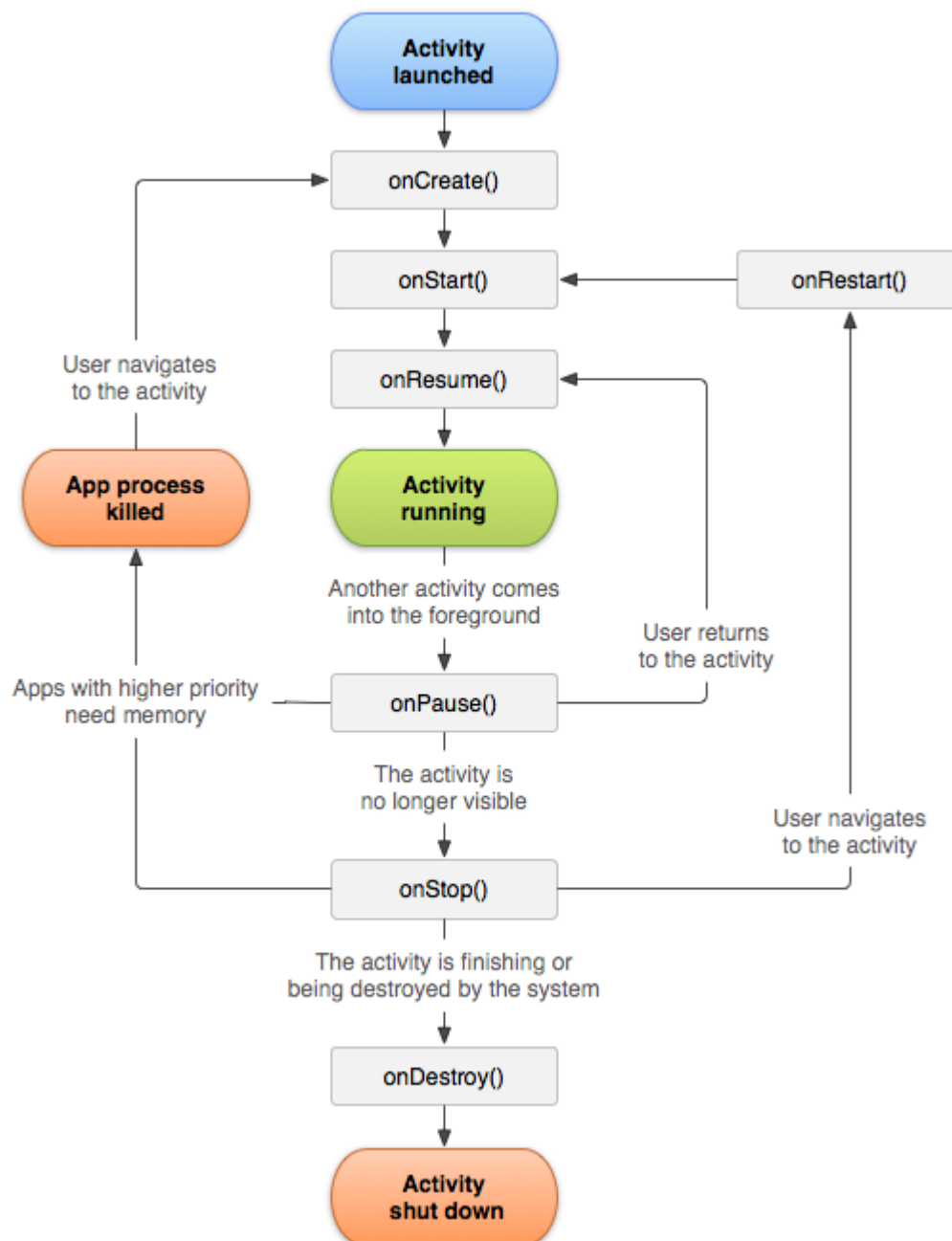


Figura 13: Ciclo de vida de una actividad

OnCreate: Éste método es llamado al crear una actividad. Se usa principalmente para configurar la pantalla y para otro tipo de configuraciones iniciales.

OnRestart: Éste método es llamado después de que una actividad se haya detenido, justo antes del método *OnStart*. Se usa principalmente para todos aquellos procesos que los métodos *OnPause* y *OnStop* no han sido capaces de acabar y es necesario acabarlos antes del método *OnStart*.

OnStart: Éste método es llamado cuando la actividad está a punto de ser visible al usuario. Se usa principalmente para refrescar valores y para que puedan ser cambiados antes de que los vea el usuario.

OnResume: Éste método es llamado cuando la actividad vuelve de un segundo plano. Se usa principalmente para realizar cualquier acción después de la interacción del usuario en el segundo plano.

OnPause: Éste método es llamado cuando la actividad se pone en segundo plano porque alguna ventana emergente o de información tape la actividad del primer plano. Se usa principalmente para confirmar cambios o destrucciones de datos.

OnStop: Éste método es llamado cuando la actividad se pone en segundo plano porque ha sido minimizada y ya no es visible para el usuario. Se usa principalmente para realizar ciertas tareas antes de la destrucción de la actividad o para prepararse para su reconstrucción.

OnDestroy: Éste método es llamado cuando justo antes de que la actividad sea destruida. Se usa principalmente como medio final para guardar los últimos datos.

4.3.3 Carpeta resources

Ésta carpeta contiene todos los recursos necesarios para el proyecto tales como: imágenes, layouts (donde se encuentran los archivos xml), estilos, etc..

Los tipos de recursos se distribuyen a su vez en las siguientes subcarpetas:

Carpeta	Descripción
/res/drawable/	<p>Contiene las imágenes y otros elementos gráficos usados por la aplicación. Para poder definir diferentes recursos dependiendo de la resolución y densidad de la pantalla del dispositivo se suele dividir en varias subcarpetas:</p> <ul style="list-style-type: none"> • /drawable (recursos independientes de la densidad) • /drawable-ldpi (densidad baja) • /drawable-mdpi (densidad media) • /drawable-hdpi (densidad alta) • /drawable-xhdpi (densidad muy alta) • /drawable-xxhdpi (densidad muy muy alta :)
/res/mipmap/	<p>Contiene los iconos de lanzamiento de la aplicación (el icono que aparecerá en el menú de aplicaciones del dispositivo) para las distintas densidades de pantalla existentes. Al igual que en el caso de las carpetas /drawable, se dividirá en varias subcarpetas dependiendo de la densidad de pantalla:</p> <ul style="list-style-type: none"> • /mipmap-mdpi • /mipmap-hdpi • /mipmap-xhdpi • ...
/res/layout/	<p>Contiene los ficheros de definición XML de las diferentes pantallas de la interfaz gráfica. Para definir distintos <i>layouts</i> dependiendo de la orientación del dispositivo se puede dividir también en subcarpetas:</p> <ul style="list-style-type: none"> • /layout (vertical) • /layout-land (horizontal)
/res/anim/ /res/animator/	Contienen la definición de las animaciones utilizadas por la aplicación.
/res/color/	Contiene ficheros XML de definición de listas de colores según estado.
/res/menu/	Contiene la definición XML de los menús de la aplicación.
/res/xml/	Contiene otros ficheros XML de datos utilizados por la aplicación.
/res/raw/	Contiene recursos adicionales, normalmente en formato distinto a XML, que no se incluyan en el resto de carpetas de recursos.
/res/values/	Contiene otros ficheros XML de recursos de la aplicación, como por ejemplo cadenas de texto (<i>strings.xml</i>), estilos (<i>styles.xml</i>), colores (<i>colors.xml</i>), arrays de valores (<i>arrays.xml</i>), tamaños (<i>dimens.xml</i>), etc.

Tabla 2: Subcarpetas de la carpeta resources

4.3.4 Carpeta Gradle Scripts

Esta carpeta está formada por un conjunto de ficheros que permiten la compilación y agrupamiento de la aplicación en un *apk* para que pueda ser ejecutable. La función principal de esta carpeta es hacer sencillo para el usuario la parte más complicada de la programación en Android para que de esta manera la curva de aprendizaje de la programación de aplicaciones sea menor y poder llevar así la programación Android a un mayor público.

4.4 Partes de la aplicación

A continuación se detallarán las partes en las que se divide la aplicación y se explicarán con detalle tanto sus elementos como su desarrollo.

4.4.1 Pantalla principal



Figura 14: Pantalla principal

Esta pantalla es la que da la bienvenida, se encuentra en el primer nivel de la aplicación y es la primera en aparecer al abrir la aplicación.

Esta pantalla es una de las más simple de todas y está formada por 4 botones los cuales accederán a las 4 funciones principales de la aplicación. Además, al iniciar la aplicación por primera vez pedirá al usuario permisos para acceder a la ubicación, sin los cuales la cuantificación del ejercicio que se hará más adelante no podría hacerse.

4.4.2 Pantalla perfiles

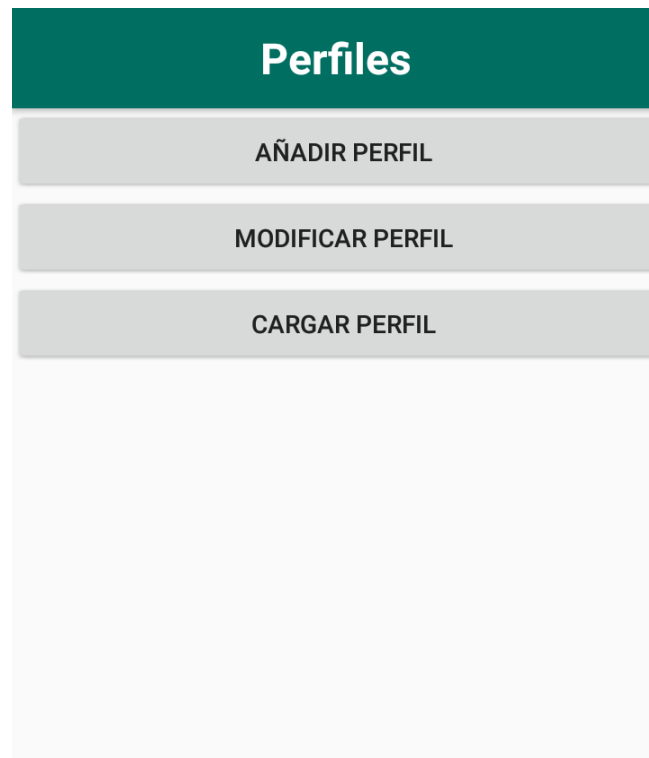


Figura 15: Pantalla perfiles

Esta pantalla se encuentra en el segundo nivel y da acceso a la primera de las funciones de la aplicación, específicamente a todas las referentes a los perfiles.

Esta pantalla es, junto a la principal, la más simple de la aplicación. Únicamente está formado por 3 botones que nos llevarán a 3 funciones distintas dentro de perfiles.

4.4.2.1 Pantalla añadir perfil



The screenshot shows a mobile application screen titled "Añadir perfil". The screen contains the following elements from top to bottom:

- A dark green header bar with the text "Añadir perfil" in white.
- A text input field labeled "Email".
- A text input field labeled "Contraseña".
- A section labeled "Sexo:" containing two radio button options: "Masculino" and "Femenino".
- A text input field labeled "Altura (cm)".
- A text input field labeled "Peso (kg)".
- A text input field labeled "Edad".
- A grey button labeled "REGISTRAR".

Figura 16: Pantalla Añadir perfil

Esta pantalla se encuentra en el tercer nivel y permite insertar un nuevo perfil a la base de datos. Está formada por los siguientes elementos:

EditText email: Aquí el usuario escribirá el nuevo email, deberá mantener una estructura correcta (ejemplo@ejemplo.com) ya que en caso contrario no sería aceptado y se notificaría de ello al usuario.

EditText contraseña: Aquí el usuario escribirá la nueva contraseña, deberá tener un tamaño de al menos 6 caracteres, en caso contrario no sería aceptada y se notificaría de ello al usuario.

RadioGroup sexo: En este caso no hace falta escribir nada, únicamente elegir la opción que corresponde al sexo del usuario.

EditText altura: Aquí el usuario escribirá en centímetros la altura del nuevo perfil, en caso de que sea mayor de 280 cm (altura del ser humano más alto de la historia) no sería aceptado y se notificaría de ello al usuario.

EditText peso: Aquí el usuario escribirá en kilogramos el peso del nuevo perfil, en caso de que sea mayor de 150 (máximo peso aconsejable para realizar ejercicio) no sería aceptado y se notificaría de ello al usuario.

EditText edad: Aquí el usuario escribirá la edad del nuevo perfil, en caso de que sea mayor de 110 (con el fin de evitar perfiles no realistas) no sería aceptado y se notificaría de ello al usuario.

Botón registrar: Este botón es el que se encarga (una vez pulsado) de comprobar que todos los campos anteriores no estén vacíos y de que los requisitos antes mencionados se hayan cumplido correctamente, si no fuese así lo notificaría al usuario. Una vez revisado lo ya comentado iniciará y abrirá la base de datos de los perfiles y comprobará si el email elegido se encuentra ya registrado, en caso de que ya se encuentre registrado este no sería aceptado y se notificaría de ello al usuario, en caso de que no se encontrase ya registrado se registrarían todos los datos y se notificaría de ello al usuario.

4.4.2.2 Pantalla modificar perfil

The screenshot shows a mobile application interface for modifying a profile. The title bar is dark green with the text 'Modificar perfil' in white. Below the title bar, there are several input fields with light gray borders and placeholder text: 'Email', 'Contraseña', 'Contraseña', 'Altura (cm)', 'Peso (kg)', and 'Edad'. There are also two radio buttons for 'Masculino' and 'Femenino'. At the bottom, there are three buttons: 'BUSCAR', 'MODIFICAR', and 'ELIMINAR'.

Figura 17: Pantalla modificar perfil

Esta pantalla se encuentra en el tercer nivel y permite modificar un perfil de la base de datos. Está formada por los siguientes elementos:

EditText email: Aquí el usuario escribirá el email del perfil que quiera modificar, al igual que en el caso anterior, el correo deberá mantener una estructura correcta (ejemplo@ejemplo.com) ya que en caso contrario no sería aceptado y se notificaría de ello al usuario.

EditText contraseña: Aquí el usuario escribirá la contraseña del perfil que quiera modificar, al igual que en el caso anterior, la contraseña deberá tener un tamaño de al menos 6 caracteres, en caso contrario no sería aceptado y se notificaría de ello al usuario.

Botón buscar: Este botón es el que se encarga (una vez pulsado) de comprobar que los dos campos anteriores no estén en blanco y de que todos los requisitos antes mencionados se hayan cumplidos. Si no es así se notificaría de ello al usuario y si sí los cumple comprobará si coincide el email y la contraseña introducidos en la base de datos de perfiles notificando en este caso si encuentra alguno (sí está registrado) o si no ha encontrado ninguno (no está registrado).

Una vez encontrado el perfil con éxito se habilitarán los demás elementos de la pantalla y se rellenarán con los datos guardados del perfil cargado antes.

EditText contraseña: Aquí se escribirá la contraseña asociada al perfil cargado. El usuario podrá borrarla y cambiarla por otra o mantenerla. Igual que en los casos anteriores la contraseña deberá tener un tamaño de al menos 6 caracteres, en caso contrario no sería aceptado y se notificaría de ello al usuario

RadioGroup sexo: Aquí se marcará la opción asociada con el perfil cargado y el usuario podrá elegir entre mantenerla o cambiarla.

EditText altura: Aquí se escribirá la altura asociada al perfil cargado. El usuario podrá borrarla y cambiarla por otra o mantenerla. Igual que en el caso anterior (añadir perfil) en caso de que sea mayor de 280 cm no sería aceptado y se notificaría de ello al usuario.

EditText peso: Aquí se escribirá el peso asociado al perfil cargado. El usuario podrá borrarlo y cambiarlo por otro o mantenerlo. Igual que en el caso anterior (Añadir perfil) en caso de que sea mayor a 150 kg no sería aceptado y se notificaría de ello al usuario.

EditText edad: Aquí se escribirá la edad asociada al perfil cargado. El usuario podrá borrarla y cambiarla o mantenerla. Igual que en el caso anterior (Añadir perfil) en caso de que sea mayor de 110 años no sería aceptada y se notificaría de ello al usuario.

Botón modificar: Este botón es el que se encarga (una vez haya sido pulsado) de comprobar que todos los campos que van después del botón buscar no se encuentren vacíos y de que todas las condiciones antes descritas se hayan cumplido, si no fuese así lo notificaría al usuario. Si está todo correcto se abriría la base de datos de los perfiles, se volvería a encontrar el email antes cargado y se actualizarían todos los datos nuevos, después se deshabilitarían todos los elementos que existen por debajo del botón buscar a la espera de que se cargue otro perfil correcto en los dos primeros campos de la pantalla.

Botón Eliminar: Este botón es el que se encarga (una vez haya sido pulsado) de eliminar el perfil antes cargado. Como la función de buscar si el email y la contraseña cargados existen en la base de datos de perfiles ya lo ha hecho el botón buscar no se necesita realizar ninguna comprobación por lo que todos los datos del perfil de la base de datos se eliminarían y se deshabilitarían todos los elementos que existen por debajo del botón buscar a la espera de que se cargue otro perfil correcto en los dos primeros campos de la pantalla.

4.4.2.3 Pantalla cargar perfil



Figura 18: Pantalla cargar perfil

Esta pantalla se encuentra en el tercer nivel y permite cargar un perfil. El perfil cargado será el usado en la realización de los cálculos de la cuantificación de ejercicio. Está formada por los siguientes elementos:

EditText email: De la misma manera que en los dos casos anteriores aquí el usuario escribirá el email del perfil que quiera modificar, el correo deberá mantener una estructura correcta (ejemplo@ejemplo.com) ya que en caso contrario no sería aceptado y se notificaría de ello al usuario.

EditText contraseña: De la misma manera que en los dos casos anteriores aquí el usuario escribiría la contraseña del perfil que queremos modificar, la contraseña deberá tener un tamaño de al menos 6 caracteres, en caso contrario no sería aceptado y se notificaría de ello al usuario.

Botón cargar: En este caso este botón tendrá una función parecida a la del *botón buscar* pero no completamente igual. Una vez pulsado se comprobará en la base de datos de los perfiles si el email y la contraseña se encuentran, en caso de no ser así se notificará de ello al usuario y en caso de encontrarlo el email pasaría a la base de datos load, esta base de datos únicamente contendrá una fila y una columna en el que se escribirá el correo que haya sido cargado.

4.4.3 Pantalla ejercicio

Figura 19: Pantalla ejercicio

Esta pantalla es la más importante de la aplicación ya que todas las anteriores recopilaban información para volcarla en esta. En esta pantalla además se realiza la cuantificación física que permitirá realizar los avisos nutricionales y guardar los datos en el historial, por lo que esta pantalla es además fundamental para que la pantalla historial haga su función. Los elementos que la forman son los siguientes:

PlainText espacio: Este plainText sólo hará su función si no existe cargado ningún perfil, en ese caso aparecerá el siguiente texto: *"No hay cargado ningún perfil, por lo que algunos campos no serán posibles calcularlos y aparecerán vacíos"*. En el caso de sí existir un perfil cargado no aparecerá ningún texto y el plainText simplemente funcionará como un espacio entre el límite superior de la pantalla ejercicio y el siguiente elemento (el spinner).

Spinner: Al pulsar sobre este elemento se despliega una columna con todos los ejercicios disponibles que la aplicación es capaz de cuantificar, el usuario deberá elegir uno para poder continuar ya que hasta que no se haya elegido uno el *botón comienzo* se encontrará deshabilitado. Los ejercicios a elegir son : caminar a ritmo normal, caminar a ritmo rápido, correr a ritmo normal, correr a ritmo rápido, bicicleta y remo. De entre estos ejercicios los ejercicios aeróbicos son los 3 primeros y los anaeróbicos los 3 últimos.

Botón comienzo: Este botón es el que permitirá comenzar la cuantificación del ejercicio. Al pulsarlo se comprobará que la señal GPS esté activada, si no fuera así se notificaría al usuario y se mostraría la pantalla de los ajustes que permite activar el GPS. Si ya estuviera activado, el control de distancia por GPS comenzaría mediante el uso de la variable LocationManager y su correspondiente listener, de esta forma cada vez que haya pasado 1 segundo y se haya detectado que se ha avanzado al menos 1 metro los campos que ofrecen

la información del ejercicio se actualizarán. Además, cada vez que este botón es pulsado todos los campos se inicializarán a cero.

Botón fin: Este botón es el que finalizará la cuantificación del ejercicio. Al pulsarse este botón la variable `LocationManager` dejará de realizar su función y ya no detectará movimiento, si existe un perfil cargado le ofrecerá al usuario la posibilidad de guardar dicho ejercicio en su historial, en caso contrario haría nada. Si en la pantalla alarmas se marcó la opción de mostrar avisos nutricionales ejecutará la parte del código que esperará el tiempo necesario antes de mostrar la notificación.

Cronómetro: Este elemento simplemente mostrará el tiempo transcurrido desde que el usuario pulsó el botón comienzo hasta que pulsó el botón fin. Sin embargo, de una manera oculta para el usuario, se contabiliza el tiempo real que el ejercicio ocupa el cual sí será visible en la pantalla historial si el usuario decide guardar este ejercicio en su historial.

PlainText metros: Este elemento únicamente sirve para dar título al campo en el que aparecerán los metros recorridos.

EditText metros: Se calcula gracias a una la función del `locationManager` antes descrita y acumulando la velocidad instantánea (que son los metros recorridos en un segundo). Este elemento puede actuar de 3 posibles maneras:

1. Si en alarmas la opción *mostrar metros* no está marcada: En este caso la contabilización de metros no se haría y en este campo se mostraría un guión.
2. Si en alarmas la opción *mostrar metros* está marcada y no hay ningún perfil cargado: En este caso la contabilización de metros sí se haría y en este campo se mostrarían los metros recorridos.
3. Si en alarmas la opción *mostrar metros* está marcada y sí hay un perfil cargado: En este caso la contabilización de metros sí se haría y en este campo se mostrarían los metros recorridos.

PlainText velocidad instantánea: Este elemento únicamente sirve para dar título al campo en el que aparecerá la velocidad instantánea.

EditText velocidad instantánea: Se calcula gracias a una de las funciones del `locationManager`. Este elemento puede actuar de 3 posibles maneras:

1. Si en alarmas la opción *mostrar velocidad instantánea* no está marcada: En este caso la velocidad instantánea no se mostraría y en este campo se mostraría un guión.
2. Si en alarmas la opción *mostrar velocidad instantánea* está marcada y no hay ningún perfil cargado: En este caso la velocidad instantánea sí se mostraría y en este campo se mostraría la velocidad del usuario en ese instante.
3. Si en alarmas la opción *mostrar velocidad instantánea* está marcada y sí hay un perfil cargado: En este caso la velocidad instantánea sí se mostraría y en este campo se mostrarían la velocidad del usuario en ese instante.

PlainText velocidad media: Este elemento únicamente sirve para dar título al campo en el que aparecerá la velocidad media.

EditText velocidad media: Se calcula dividiendo los metros recorridos entre el tiempo que ha durado el ejercicio. Este elemento puede actuar de 3 posibles maneras:

1. Si en alarmas la opción *mostrar velocidad media* no está marcada: En este caso la velocidad media no se calcularía y en este campo se mostraría un guión.

2. Si en alarmas la opción *mostrar velocidad media* está marcada y no hay ningún perfil cargado: En este caso la velocidad media sí se calcularía y en este campo se mostraría la velocidad media del usuario.
3. Si en alarmas la opción *mostrar velocidad media* está marcada y sí hay un perfil cargado: En este caso la velocidad media sí se calcularía y en este campo se mostrarían la velocidad media del usuario.

PlainText kilocalorías: Este elemento únicamente sirve para dar título al campo en el que aparecerán las kilocalorías gastadas.

EditText kilocalorías: Se calcula usando la siguiente fórmula: $\text{peso(kg)} * \text{duracion del ejercicio (minutos)} * \text{MET} * 0.175$, en el caso de que fuese una mujer a la formula se le añadiría un $* 0.9$.

El MET es la unidad de medida del índice metabólico y dependiendo del ejercicio que se haga tendrá un valor u otro. Al pedir al usuario mediante el *spinner* que especificara el tipo de ejercicio que va a hacer sabremos qué valor de MET añadir. Para saber los valores de los METs se pueden consultar tablas en internet que poseen METs generalizados que permiten aproximar de una manera muy cercana el valor de las kilocalorías gastadas. Cabe destacar que cada usuario tendría un valor de MET distinto para cada ejercicio pero gracias a estas tablas podremos generalizarlo y así adaptarlo a todas las personas. La tabla usada de METs ha sido la siguiente:

Actividad	METs
Caminar ligero (4,5 km/h)	3,3
Caminar moderadamente rápido (5,3 km/h)	3,8
Caminar rápido (6,4 km/h)	5,0
Correr (8,4 km/h)	9,0
Correr (9,6 km/h)	10,0
Correr (10,8 km/h)	11,0
Correr (11,3 km/h)	11,5
Correr (12,1 Km/h)	12,5
Correr (12,9 Km/h)	13,5
Correr (13,8 Km/h)	14,0
Correr (14,5 Km/h)	15,0
Correr (16,1 Km/h)	16,0
Correr (17,5 Km/h)	18,0
Bici estática (100 watts)	5,5
Bici estática (150 watts)	7,0
Bici estática (200 watts)	10,5
Bici estática (250 watts)	12,5
Bici estática (300 watts)	18,0
Remo (50 watts)	3,5
Remo (100 watts)	7,0
Remo (150 watts)	8,5
Remo (200 watts)	12,0
Bicicleta paseo (<16 km/h)	4,0
Bicicleta (esfuerzo ligero: 16-19 km/h)	6,0
Bicicleta (esfuerzo medio: 19-22,5 km/h)	8,0
Bicicleta (esfuerzo vigoroso: 22,5-24 km/h)	10,0

Tabla 3: Tabla valores METs

Este elemento puede actuar de 3 posibles maneras:

1. Si en alarmas la opción *mostrar kilocalorías* no está marcada: En este caso las kilocalorías no se calcularían y en este campo se mostraría un guión.
2. Si en alarmas la opción *mostrar kilocalorías* está marcada y no hay ningún perfil cargado: En este caso las kilocalorías no se calcularían y este campo se mostraría vacío.
3. Si en alarmas la opción *mostrar kilocalorías* está marcada y sí hay un perfil cargado: En este caso las kilocalorías sí se calcularían y en este campo se mostrarían las kilocalorías gastadas por el usuario durante el ejercicio.

PlainText frecuencia cardiaca: Este elemento únicamente sirve para dar título al campo en el que aparecerá el pulso adecuado para la realización del ejercicio.

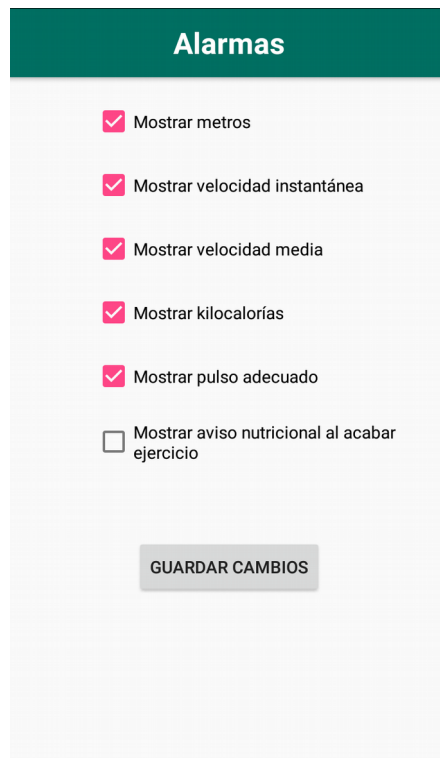
EditText frecuencia cardiaca: Se calcula usando el pulso máximo, este tendrá un valor distinto dependiendo del sexo, para las mujeres será: $214 - 0.8 * edad$ y para los hombres: $209 - 0.7 * edad$. Una vez conocido el pulso cardiaco máximo el rango de pulso adecuado para el ejercicio dependerá de la intensidad del mismo distinguiendo entre baja, media y alta, es decir:

- En el caso de un ejercicio con intensidad baja el rango de pulso adecuado tendrá como mínimo el 60% del pulso máximo y como máximo el 70% del pulso máximo.
- En el caso de un ejercicio con intensidad media el rango de pulso adecuado tendrá como mínimo el 70% del pulso máximo y como máximo el 80% del pulso máximo.
- En el caso de un ejercicio con intensidad alta el rango de pulso adecuado tendrá como mínimo el 80% del pulso máximo y como máximo el 90% del pulso máximo.

Este elemento puede actuar de 3 posibles maneras:

1. Si en alarmas la opción *mostrar pulso adecuado* no está marcada: En este caso el pulso adecuado no se calcularía y en este campo se mostraría un guión.
2. Si en alarmas la opción *mostrar pulso adecuado* está marcada y no hay ningún perfil cargado: En este caso el pulso adecuado no se calcularía y este campo se mostraría vacío.
3. Si en alarmas la opción *mostrar pulso adecuado* está marcada y sí hay un perfil cargado: En este caso el pulso adecuado sí se calcularía y en este campo se mostraría el rango de pulso adecuado para la realización de este ejercicio.

4.4.4 Pantalla alarmas



Alarmas

- ☒ Mostrar metros
- ☒ Mostrar velocidad instantánea
- ☒ Mostrar velocidad media
- ☒ Mostrar kilocalorías
- ☒ Mostrar pulso adecuado
- ☐ Mostrar aviso nutricional al acabar ejercicio

GUARDAR CAMBIOS

Figura 20: Pantalla alarmas

Esta pantalla se encuentra en el segundo nivel y permite configurar los parametros que se mostrarán durante la realización del ejercicio. Está formada por 6 radio buttons y un botón. No está asociado con la base de datos de perfiles, tiene su base de datos propias, por lo que la configuración de las alarmas se mantendrá aunque carguemos otro perfil. Al contrario que las anteriores pantallas ésta no espera al que el usuario interactúe con los elementos de la pantalla para acceder a su base de datos, en esta pantalla nada más entrar en ella se consulta la base de datos de las alarmas para comprobar si se encuentra inicializada, si no es así no haría nada y si sí se encontrase inicializada marcaría los radio buttons tal y como se encuentra guardado en la base de datos (de esta manera todos los cambios anteriores guardados por el usuario se mantendrían). Por tanto los elementos que forman esta pantalla son los siguientes:

Radio button metros: Con este elemento marcado el usuario podrá visualizar los metros que lleva recorridos en la pantalla ejercicio, los cuales serán mostrados en el campo metros de dicha actividad, en caso de no marcarse se mostrará un guion en dicho campo.

Radio button velocidad instantánea: Con este elemento marcado el usuario podrá visualizar la velocidad instantánea en la pantalla ejercicio, la cual será mostrada en el campo velocidad instantánea de dicha actividad, en caso de no marcarse se mostrará un guión en dicho campo.

Radio button velocidad media: Con este elemento marcado el usuario podrá visualizar la velocidad media en la pantalla ejercicio, la cual será mostrada en el campo velocidad media de dicha actividad, en caso de no marcarse se mostrará un guion en dicho campo.

Radio button kilocalorías: Con este elemento marcado el usuario podrá visualizar las kilocalorías en la pantalla ejercicio, las cuales serán mostrados en el campo kilocalorías de dicha actividad, en caso de no marcarse se mostrará un guion en dicho campo.

Radio button pulso adecuado: Con este elemento marcado el usuario podrá visualizar el pulso adecuado para la realización del ejercicio en la pantalla ejercicio, el cual será mostrado en el campo pulso adecuado de dicha actividad, en caso de no marcarse se mostrará un guion en dicho campo.

Radio button aviso nutricional: Con este elemento marcado el usuario podrá recibir un aviso nutricional al acabar el ejercicio en la pantalla ejercicio, este aviso dependerá del tiempo empleado en el ejercicio y del tipo de ejercicio que se haya realizado.

Botón guardar cambios: Este botón, al ser pulsado se encargará de actualizar la base de datos de las alarmas teniendo en cuenta los radio button marcados. Lo único que tendrá en cuenta a la hora de realizar esta acción es si la base de datos no ha sido inicializada ya que, en ese caso, los insertará y en caso de ya estar inicializada los actualizará.

4.4.5 Pantalla historial

Historial				
Su TMB (calorias quemadas en reposo) es de: 1620,3 kilocalorías				
Su IMC (Indice de masa corporal) es de: 21,9 -> Peso normal				
Día	Tipo de ejercicio	Duración (minutos)	Metros	V

Figura 21: Pantalla historial

Esta pantalla se encuentra en el segundo nivel y permite al usuario acceder a su datos personales como su historial de ejercicios, su TMB y su IMC.

Esta pantalla es la única que no interactúa con el usuario ya que solamente le muestra información y la única con orientación horizontal debido al tamaño de la tabla. Está formado por:

PlainText TMB: En este PlainText se escribirá la frase que se puede ver en la figura con las kilocalorías gastadas en reposo calculadas con la siguiente fórmula:

- Si el género es femenino: $10 * \text{peso (kg)} + 6,25 * \text{altura (cm)} - 5 * \text{edad (años)} - 161$.
- Si el género es masculino: $10 * \text{peso (kg)} + 6,25 * \text{altura (cm)} - 5 * \text{edad (años)} + 5$.

Y el estado del usuario calculado de la siguiente manera:

- Si $\text{IMC} > 18$ el estado será *bajo peso*.
- Si $\text{IMC} \geq 18$ y < 25 el estado será *peso normal*.
- Si $\text{IMC} \geq 25$ y < 27 el estado será *sobrepeso*.
- Si $\text{IMC} > 27$ el estado será *obesidad*.

PlainText IMC: En este PlainText se escribirá la frase que se puede ver en la figura con el IMC calculado con la siguiente fórmula: $\text{peso(kg)}/\text{altura}^2(\text{m})$

Tabla: En la tabla que se ve en la figura se irán añadiendo todos los ejercicios guardados en la base de datos historial asociados al correo cargado. La tabla posee en scroll horizontal y otro vertical ya que debido a su gran tamaño.

5 Integración, pruebas y resultados

5.1 Integración

Se decidió desde un primer momento que la aplicación no se iba a publicar al PS una vez estuviese acabada y así se hizo finalmente. De esta forma, el único método de integración de la aplicación a un dispositivo móvil era usando Android Studio como intermediario. Para esta operación existían dos métodos:

Método 1: Usando el dispositivo móvil como emulador: Este fue el método usado. Mediante este método no hizo falta utilizar el simulador virtual del propio Android Studio y se mejoró en rapidez de ejecución, otra ventaja de realizar así la emulación fue que en todo momento se podía comprobar cómo quedaría la aplicación y cómo respondería en un dispositivo físico y la posibilidad de debuggear la aplicación en el dispositivo. Para poder utilizar el dispositivo físico como emulador fue necesario instalar un driver específico del modelo y marca del dispositivo físico. Este driver se puede encontrar en internet y dependiendo del modelo y de la marca del móvil este driver es más o menos accesible siendo para algunos dispositivos muy fácil de encontrar y para otros realmente complicado.

Método 2: Generando el apk para luego transferirlo al dispositivo y ejecutarlo desde ahí. Este método es efectivo para instalar la aplicación en un dispositivo del cual no se tenga su driver instalado y del cual solo se busque la instalación de la aplicación, para ese caso este método sería el más rápido. Sin embargo, si lo que buscamos es realizar pruebas y por tanto reinstalar varias veces la aplicación este método resulta mucho más largo y tedioso. Además de que no permite debuggear la aplicación como sí permitía el método anterior.

La integración se realizó en 2 dispositivos móviles distintos, uno en un dispositivo de 5 pulgadas y otro en un dispositivo de 3.5 pulgadas. En el dispositivo de 5 pulgadas la integración fue perfecta, pero en el de 3.5 pulgadas es necesario el uso del scroll integrado en la pantalla para poder ver todos los elementos que forman la pantalla. A continuación se muestra una figura con la integración de ambos dispositivos.

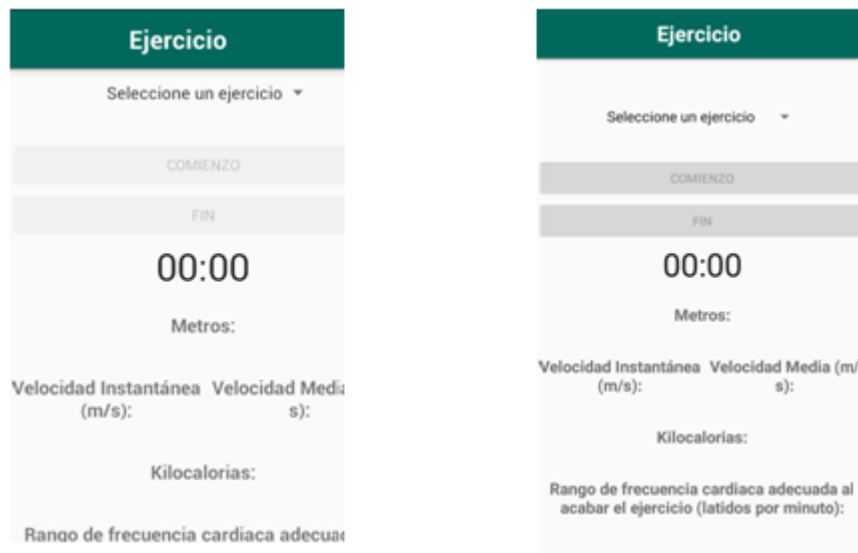


Figura 22: Integración en dispositivo móvil de 3.5 pulgadas vs integración en dispositivo móvil de 5 pulgadas

5.2 Pruebas y resultados

Se han realizado un total de 4 pruebas de la aplicación utilizando 2 perfiles distintos, es decir, 2 pruebas por perfil. De esta forma las pruebas han sido:

5.2.1 Prueba 1

5.2.1.1 Perfil

Figura 23: Prueba 1. Perfil

Los datos del primer perfil (prueba 1) son los que se ven en la figura. Corresponden por tanto a un hombre de 180 cm de altura, 75 kg de peso y 25 años de edad.

5.2.1.2 Ejercicio aeróbico



Figura 24: Prueba 1. Ejercicio aeróbico

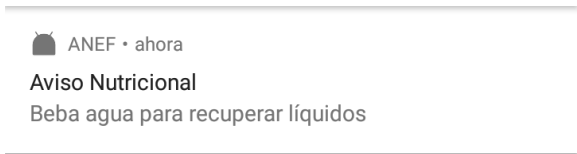


Figura 25: Prueba 1. Notificación del ejercicio aeróbico

En la primera figura se puede ver la cuantización realizada por la aplicación de la actividad aeróbica especificada en la misma. Los datos obtenidos como se pueden ver son completamente viables.

En la segunda figura se puede ver la notificación arrojada por la aplicación al acabar el ejercicio. Al ser un ejercicio aeróbico la notificación saltará al momento de darle al botón fin.

5.2.1.3 Ejercicio anaeróbico

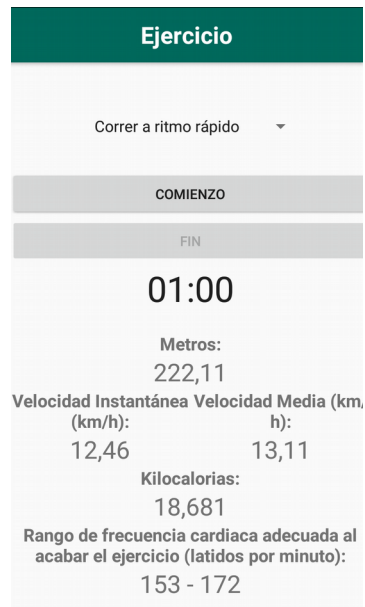


Figura 26: Prueba 1. Ejercicio anaeróbico

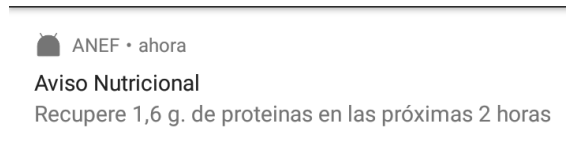


Figura 27: Prueba 1. Notificación del ejercicio anaeróbico

En la primera figura se puede ver la cuantización realizada por la aplicación de la actividad anaeróbica especificada en la misma. Los datos obtenidos como se pueden ver son completamente viables.

En la segunda figura se puede ver la notificación arrojada por la aplicación al acabar el ejercicio. Al ser un ejercicio anaeróbico la notificación saltará a los 10 minutos de darle al botón fin.

5.2.1.4 Historial

Historial		
Su TMB (calorias quemadas en reposo) es de: 1589,0 kilocalorías		
Su IMC (Indice de masa corporal) es de: 23,1 -> Peso normal		
Día	Tipo de ejercicio	Duración (minutos)
2018-06-20	Caminar a ritmo normal	2,9
2018-06-20	Correr a ritmo rápido	1,0

Figura 28: Prueba 1. Historial parte 1

Historial			
Su TMB (calorias quemadas en reposo) es de: 1589,0 kilocalorías			
Su IMC (Indice de masa corporal) es de: 23,1 -> Peso normal			
ón (minutos)	Metros	Vel. media (km/h)	Kilocalorias
2,9	213,9	4,5	12,5
1,0	222,1	13,1	18,7

Figura 29: Prueba 1. Historial parte 2

Las dos figuras corresponden a la misma tabla, solo que al ser tan grande no se ve entera en la pantalla del móvil. En ella se pueden ver los datos recopilados de los dos ejercicios antes realizados.

5.2.2 Prueba 2

5.2.2.1 Perfil

Añadir perfil

prueba2@gmail.com

.....

Sexo:

☐ Masculino

☒ Femenino

170

60

27

REGISTRAR

Figura 30: Prueba 2. Perfil

Los datos del segundo perfil (prueba 2) son los que se ven en la figura. Corresponden por tanto a una mujer de 170 cm de altura, 60 kg de peso y 27 años de edad.

5.2.2.2 Ejercicio aeróbico

Ejercicio

Caminar a ritmo normal ▼

COMIENZO

FIN

03:14

Metros:
243,17

Velocidad Instantánea (km/h): 6,30 Velocidad Media (km/h): 4,81

Kilocalorías:
10,248

Rango de frecuencia cardiaca adecuada al acabar el ejercicio (latidos por minuto):
115 - 135

Figura 31: Prueba 2. Ejercicio aeróbico

ANEF • ahora

Aviso Nutricional
Beba agua para recuperar líquidos

Figura 32: Prueba 2. Notificación del ejercicio aeróbico

Al igual que en la prueba anterior, la primera figura corresponde con la cuantización del ejercicio y la segunda con la notificación.

Como se puede observar, pese a que el tiempo ha sido el mismo y los metros recorridos parecidos el número de kilocalorías quemadas ha sido muy distinto, esto se debe al cambio de perfil y es que dependiendo de cada persona la quema de calorías en los ejercicios es distinta.

5.2.2.3 Ejercicio anaeróbico



Figura 33: Prueba 2. Ejercicio anaeróbico

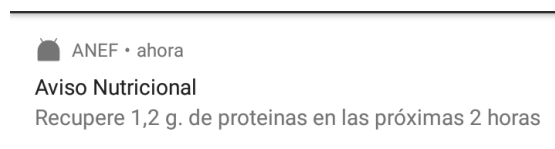


Figura 34: Prueba 2. Notificación del ejercicio anaeróbico

Al igual que en el ejercicio aeróbico de esta misma prueba las diferencias en la quema de calorías se deben al cambio de perfil. También se puede ver cómo se ve afectado en la notificación la cantidad de proteínas a recuperar tras acabar el ejercicio.

5.2.2.4 Historial

Historial		
Su TMB (calorias quemadas en reposo) es de: 1582,5 kilocalorías		
Su IMC (Indice de masa corporal) es de: 22,5 -> Peso normal		
Día	Tipo de ejercicio	Duración (minutos)
2018-06-20	Caminar a ritmo normal	3,0
2018-06-20	Correr a ritmo rápido	1,0

Figura 35: Prueba 2. Historial parte 1

Historial			
Su TMB (calorias quemadas en reposo) es de: 1582,5 kilocalorías			
Su IMC (Indice de masa corporal) es de: 22,5 -> Peso normal			
ón (minutos)	Metros	Vel. media (km/h)	Kilocalorias
3,0	243,2	4,8	10,2
1,0	211,8	12,5	14,6

Figura 36: Prueba 2. Historial parte 2

Al igual que en la prueba anterior en el historial se pueden ver recopiladas los dos ejercicios físicos realizados.

6 Conclusiones y trabajo futuro

6.1 Conclusiones

Una vez terminado el proyecto se puede decir que se han cumplido todos los objetivos propuestos al principio e incluso algunos que no se habían previsto.

Entre los objetivos cumplidos diferenciaría dos, los requisitos cumplidos por la aplicación y los conocimientos adquiridos.

Entre los requisitos cumplidos los más importantes han sido:

- Creación de las base de datos.
- Comunicación fluida con la base de datos.
- Conexión entre todas las actividades de la aplicación.
- Creación de tareas que funcionasen en segundo plano.
- Utilización de sensores del dispositivo móvil.
- Creación de notificaciones.

Entre los conocimientos adquiridos los más importantes han sido:

- Lenguaje xml.
- Leguaje Android.
- Creación de servicios.
- Uso de sensores a través del lenguaje Android.
- Conocimientos nutricionales y de actividad física.
- Bases de datos.
- Manejo del programa Android Studio.
- Conexión bluetooth.

6.2 Trabajo futuro

Con el objetivo de mejorar la aplicación ya creada y hacerla más sólida y competitiva frente a otras aplicaciones similares del mercado se proponen las siguientes ideas:

- Sincronización con pulsera de actividad vía bluetooth para la realización de la cuantificación del ejercicio. Cabe destacar que en un primer momento esta era una de las ideas del proyecto pero finalmente se descartó y se decidió hacer la cuantificación del ejercicio mediante el sensor GPS del móvil ya que por tiempo y recursos acabó siendo inviable.
- Mejorar la interfaz.
- Publicar la aplicación en Play Store.
- Aumento de los ejercicios físicos a los que se puede cuantificar el ejercicio.
- Adaptar la aplicación a distintos idiomas.
- Usar el sensor de pasos para aquellos momentos en los que la señal GPS sea muy débil (en caso de no usar actividad de pulsera).
- Optimización de aplicación de cara a ahorrar batería.
- Implementar objetivos de quema de calorías diarias.
- Adaptar la aplicación a otros sistemas operativos como IOS o Windows Phone.
- Poder eliminar filas en la tabla de la pantalla historial.

Referencias

- [1] Bill Philipps, Chris Stewart, Brian Hardy y Kristin Marsicano. "Programación con Android". ANAYA. 2016
- [2] Joan Ribas Lequerica. "Desarrollo de aplicaciones para Android". ANAYA. 2017
- [3] Jesús Tomás Gironés. "El gran libro de Android". marcombo. 2013
- [4] http://unestudiantedeinformatica.blogspot.com/2014/06/5-libros-para-aprender-programar-en_19.html
- [5] <https://developer.android.com/?hl=es-419>
- [6] <https://androidstudiofaqs.com/>
- [7] <https://www.w3schools.com/>
- [8] <https://www.realfitness.es/calculadoras/aprende-utilizar-tablas-met-calcular-calorias-quemadas/>
- [9] http://www.noalaobesidad.df.gob.mx/index.php?option=com_content&view=article&id=52&Itemid
- [10] <https://www.axahealthkeeper.com/blog/que-es-y-como-calcular-la-tasa-metabolica-basal/>

Glosario

API	Application Programming Interface
APK	Android Application Package
EPS	Escuela Politécnica Superior
UAM	Universidad Autónoma de Madrid
SO	Sistema Operativo
PS	Play Store